REMARKS

In response to the Office Action dated September 24, 2004, Applicant respectfully requests reconsideration based on the above claim amendments and the following remarks. Applicant respectfully submits that the claims as presented are in condition for allowance.

Claims 1-12 are pending in this application. Claims 1, 2, 7, and 10-12 are amended. Claims 1, 2, 7, and 10-12 contain no new matter and are supported by the original application, including the drawings and the original claims.

Claims 1-6 and 10-12 were rejected under 35 U.S.C. § 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which Applicant regards as the invention.

Claims 1, 2, 7, and 10-12 are amended. Claims 1 and 7 are amended to provide antecedent basis for "the SSL daemon process". Claim 7 is amended to correct method claim format so that each element begins with a gerund. Claims 1, 2, 10, 11, and 12 are amended to clarify the subject matter that Applicant regards as the invention.

Claim 1 was rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent No. 5,657,390 to Elgamal et al. ("Elgamal") in view of an online glossary definition of the term "Stunnel" by Trojnara et al. ("Stunnel").

A *prima facie case* of obviousness under 35 U.S.C. § 103(a) requires that the combination of references teach or suggest all the claim elements. The proposed combination of Elgamal and Stunnel in the Office Action fails to teach or suggest all the claim elements, because it fails to teach or suggest, for example, an application process using SSL API calls to communicate with an SSL wrapper process.

The Office Action states "Elgamal does not teach the SSL wrapper process." (Office Action, page 3, para. 5). Furthermore, Stunnel fails to teach or suggest an application process using SSL API calls to communicate with an SSL wrapper process. For at least these reasons, claim 1 is patentable over the combination of Elgamal and Stunnel, as discussed below.

Claim 1 recites, *inter alia*, "a plurality of SSL application programming interface (API) calls for communication between the application process and SSL wrapper

POU920010131US1
I32-0010   5

BEST AVAILABLE COPY

process". The claimed invention is very different from Stunnel. Stunnel states "The concept is that having non-SSL aware daemons running on your system you can easily setup them to communicate with clients over secure SSL channel." (Stunnel, page 1, description section, second sentence). In other words, Stunnel allows regular server applications, i.e., servers using straight socket APIs rather than SSL APIs, to connect to a local proxy that communicates with remote clients using SSL. As a result, the server application has no idea that SSL is being used. In the claimed invention, the local server applications are SSL aware, because the application process uses SSL API calls to communicate with an SSL wrapper process.

Stunnel uses similar terminology, which is probably a cause of confusion. Stunnel uses the term "SSL encryption wrapper" for something that intercepts socket APIs issued by non-SSL aware applications and then generates SSL APIs that are issued. The so-called "SSL encryption wrapper" of Stunnel acts as a proxy that neither the client or server application knows exists. Communication in and out of the server application is standard sockets in Stunnel. Communication between the server node and the remote client is standard SSL in Stunnel. Conceptually, there is a standard socket between the server application and the "SSL encryption wrapper" running in the server and a standard SSL session between the "SSL encryption wrapper" and the remote client. Any data that the "SSL encryption wrapper" receives over the socket is sent over the SSL session and vice-versa. Stunnel is simply one of many ways of implementing an SSL proxy that enables SSL to be used by application that have no knowledge of SSL. In the claimed invention, by contrast, the application process is aware of the SSL session and issues SSL APIs (not socket APIs). Stunnel fails to disclose SSL-aware applications. Therefore, claim 1 is patentable over the combination of Elgamal and Stunnel.

Claims 7, 8, 10-12 were rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent No. 6,772,333 B1 to Brendel et al. ("Brendel") in view of Stunnel.

A *prima facie case* of obviousness under 35 U.S.C. § 103(a) requires that the combination of references teach or suggest all the claim elements. The proposed combination of Brendel and Stunnel in the Office Action fails to teach or suggest all the claim elements, because it fails to teach or suggest, for example, a shared SSL session.

For at least these reasons, claim 1 is patentable over the combination of Brendel and Stunnel, as discussed below.

The Office Action states "Brendel does not teach the SSL wrapper for the process...." (Office Action, page 4, para. 9). Furthermore, Stunnel fails to teach or suggest a shared SSL session.

Claim 7 recites, *inter alia*, "a shared SSL session". In contrast to the disclosure in Stunnel, a shared SSL session is not tied to one specific application process. By definition, a shared session is used by multiple application processes concurrently. A shared SSL session is not owned by any application process. Instead, it is owned by a separate process, the SSL daemon process. Application processes issue SSL APIs to communicate over the SSL session with remote clients. For unshared SSL sessions, the SSL APIs are processed within the application process. For shared SSL sessions, the SSL APIs are passed to the SSL daemon process to be processed. This is a major difference between the claimed shared SSL sessions and SSL proxy methods, such as those disclosed in Stunnel. Stunnel fails to disclose the claimed shared SSL session. Therefore, claim 7 is patentable over the combination of Brendel and Stunnel.

Claims 8, 10, and 11 recite, *inter alia*, "a shared SSL session." As discussed above with respect to claim 7, Stunnel fails to disclose the claimed shared SSL session. Therefore, claims 8, 10, and 11 are also patentable over the combination of Brendel and Stunnel.

Claim 12 depends from claim 11 and, thus, inherits at least the patentable subject matter of claim 11. Therefore, claim 12 is also patentable over the combination of Brendel and Stunnel.

Claims 2-6 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Elgamal in view of Stunnel and further in view of Brendel.

Claims 2, 3, and 5 recite, *inter alia*, "a shared SSL session". As discussed above with respect to claim 7, Stunnel fails to disclose the claimed shared SSL session. Therefore, claims 2, 3, and 5 are also patentable over the combination of Elgamal, Brendel, and Stunnel.

Claims 4 and 6 depend, directly or indirectly, from claim 2 and, thus, inherits at least the patentable subject matter of claim 2. Therefore, claims 4 and 6 are also patentable over the combination of Brendel and Stunnel.

Claim 9 was objected to as being dependent upon a rejected base claim, but the Office Action indicated claim 9 would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.

Claim 9 depends from claim 7. For the reasons given above with respect to claim 7, Applicant believes claim 7 is allowable, and, therefore claim 9 is also allowable in dependent form.

In view of the foregoing remarks and amendments, Applicant submits that the above-identified application is now in condition for allowance. Early notification to this effect is respectfully requested.

If there are any charges with respect to this response or otherwise, please charge them to Deposit Account 09-0463 maintained by Assignee.

Respectfully submitted,

By: _Lea A. Nicholson_

Lea A. Nicholson
Registration No. 48,346
CANTOR COLBURN LLP
55 Griffin Road South
Bloomfield, CT 06002
Telephone (860) 286-2929
Facsimile (860) 286-0115
Customer No. 46429